11:2 2020

Algebraic Statistics

THE NORM OF THE SATURATION OF A BINOMIAL IDEAL, WITH APPLICATIONS TO MARKOV BASES

DAVID HOLMES

msp

Algebraic Statistics
Vol. 11, No. 2, 2020
https://doi.org/10.2140/astat.2020.11.169

THE NORM OF THE SATURATION OF A BINOMIAL IDEAL, WITH APPLICATIONS TO MARKOV BASES

DAVID HOLMES

Let B be a finite set of pure binomials in the variables x_i, and write I_B for the ideal generated by these binomials. We define a new measure of the complexity of the saturation of the ideal I_B with respect to the product of the x_i, which we call the *norm* of B. We give a bound on the norm in terms of easily computed invariants of B. We discuss statistical applications, both practical and theoretical.

1. Introduction 169
2. Examples 175
3. Computational experiments 178
4. Proof of the main results 180
Acknowledgements 186
References 186

1. Introduction

1.1. ***Background.*** Let A be a $k \times r$ matrix with integer entries, and let $u \in \mathbb{N}^r$ be a vector with nonnegative entries. The *fibre containing* u is defined as

$$\mathcal{F}(u) = \{v \in \mathbb{N}^r : Au = Av\}. \tag{1.1.1}$$

Understanding the structure of this fibre is important in a number of statistical tests. For example, the vectors in $\mathbb{N}^r$ might represent tables of data, and the matrix A might output the row and column sums of these tables, so the fibre consists of all tables with nonnegative entries and with the same row and column sums as the starting table u. See [Diaconis and Sturmfels 1998] for more details and examples. In particular, one often wants to generate samples from some probability distribution (often uniform or hypergeometric) on the fibre. If the fibre is small it is feasible to simply enumerate all the elements of the fibre. However, in practical applications the fibre is often far too large to enumerate, and the standard approach is to perform a *random walk* in the fibre, generating samples via the Metropolis–Hastings Markov chain Monte Carlo algorithm. In order to perform a random walk, we must upgrade the fibre into a graph (whose vertices are the elements of the fibre). The requirements for the Metropolis–Hastings algorithm are rather mild, the key condition is that the graph must be *connected* (since the random walk will always remain within its starting connected component).

MSC2010: 13P25, 14M25.
Keywords: markov basis, saturation, toric ideals.

1.1.1. *A random walk in the fibre.* The most naive way to convert the fibre into a graph is to choose a generating set B for the kernel $K \subseteq \mathbb{Z}^r$ of A as a $\mathbb{Z}$-module, and then form a (simple, undirected) graph by putting an edge between distinct vertices v_1 and v_2 whenever $v_1 - v_2 \in B$ or $v_2 - v_1 \in B$. We say $\mathcal{F}(u)$ is *connected by* B if the resulting graph is connected. In Section 2 we will see several examples of B that *fail* to connect $\mathcal{F}(u)$. The major innovation of Diaconis and Sturmfels [1998] was to give an algorithm to construct a generating set B which connects *every* fibre of a given matrix A.

1.1.2. *Saturated ideals and connected fibres.* To describe their result, we need a little more notation. Given $b \in B$, we write $b = b^+ - b^-$, both summands having nonnegative entries. In the ring $R = \mathbb{Z}[x_1, \ldots, x_r]$ we form the elements

$$x^{b^+} := \prod_{i=1}^{r} x_i^{b_i^+}, \quad x^{b^-} := \prod_{i=1}^{r} x_i^{b_i^-}, \tag{1.1.2}$$

and define an ideal $\mathcal{I}_B = (x^{b^+} - x^{b^-} : b \in B) \subseteq R$. Then the key theorem is the following (where we use [Sturmfels 1996, Lemma 12.2 p. 114] to interpret toric ideals as saturated ideals).

Theorem 1.1 [Diaconis and Sturmfels 1998]. *Fix a $k \times r$ matrix A, and let B be a generating set for the integral kernel of A. Suppose the ideal $\mathcal{I}_B$ is saturated with respect to the element $x_1 \cdots x_r \in R$. Then for every $u \in \mathbb{N}^r$, the fibre $\mathcal{F}(u)$ is connected by B.*

If $\mathcal{I}_B$ is saturated, B is often called a *Markov basis* (though we use the word "basis", this should not be interpreted as implying linear independence of the elements of B). The theorem then tells us that we can generate samples according to our preferred distribution by following the naive random walk algorithm above using the basis B.

On the other hand, suppose that we have a generating set B such that $\mathcal{I}_B$ is not saturated. We can (at least in principal) apply a standard saturation algorithm to $\mathcal{I}_B$ to produce a saturated ideal, and moreover the generating set produced will in fact consist of pure difference binomials (i.e. differences of monomials; see Definition 4.1). Reversing the procedure (1.1.2) we can recover a new generating set B' for the kernel K of A, and following the above theorem of Diaconis–Sturmfels, this generating set will connect all fibres, enabling efficient sampling.

Thus, when it is possible to compute this saturation, the problem is essentially solved. However, the standard algorithm for saturation involves r computations of Gröbner bases (where r is the number of columns as above), and is at present only practical for relatively small examples. General purpose algorithms (not taking advantage of the toric structure) are available in many packages (such as MAGMA [Bosma et al. 1997] and Singular [Decker et al. 2019]), and also specialised implementations for the toric case are available (CoCoa [Bigatti et al. 1999], 4ti2 [Hemmecke et al. 2001–]).

1.1.3. *Connected fibres without saturation.* The difficulty of computing the saturation motivated Aoki, Hara and Takemura [Hara et al. 2012] to suggest an algorithm for generating samples without needing to compute the saturation. They begin in the same way, with a generating set $B = \{b_1, \ldots, b_n\}$ for the integral kernel, but instead of making moves consisting of addition or subtraction of a single element of B, they instead generate n nonnegative integers a_i from a Poisson distribution with some chosen mean λ,

and n elements $\epsilon_i \in \{+1, -1\}$, and their move consists of addition of $\sum_i \epsilon_i a_i b_i$ if the result lies in the fibre, and staying put otherwise. Since the Poisson distribution generates every nonnegative integer with nonzero probability it is immediate that the resulting fibre is connected; in fact, the graph on the fibre is a complete graph, but with highly nonuniform probability of selecting moves from among edges.

They then perform a number of numerical experiments with various values of λ. In cases where it was possible to compute the saturation, they show that for careful choice of λ their algorithm performs comparably to that coming from a Markov basis, and they also illustrate that their algorithm can be applied in cases where the saturation is too hard to compute (though they can of course provide no guarantee that their algorithm is converging in reasonable time; it appears to do so, but this might be deceptive if the fibre has some connected components that are very hard to hit — see Section 1.4).

There is some tension in the use of this algorithm when it comes to choosing the value of λ. If one chooses λ very large then the algorithm takes a long time before it (appears to) converge. On the other hand, a small value of λ will product more rapid *apparent* convergence, but there is a greater risk that one is simply failing to see one or more connected components of the fibre in the time for which the algorithm is run.

1.2. *Results.*

1.2.1. *A bound on the complexity of the saturation.* In the light of the above discussion it is natural to try to bound how large and complex the saturation of the ideal $\mathcal{I}_B$ can get. To make this more precise, we define the *norm* of the generating set B as follows.

Definition 1.2. Let B be set of $n \geq 1$ vectors in $\mathbb{Z}^r$. We write $\mathcal{I}_B$ for the ideal in $R = \mathbb{Z}[x_1, \ldots, x_r]$ as defined in Section 1.1.2. The *norm* of B is the smallest integer $N \geq 1$ such that there exists a finite generating set G for the saturation of $\mathcal{I}_B$ with respect to $x_1 \cdots x_r$, with the properties that

(1) Every element of G is a pure difference binomial;

(2) Every $g \in G$ can be written in the form

$$g = \sum_{i=1}^{N} \epsilon_i m_i (x^{b_i^+} - x^{b_i^-}). \tag{1.2.1}$$

where the $\epsilon_i \in \{-1, 0, 1\}$, the m_i are Laurent monomials, and the b_i are elements of B.

The main result of this paper is the following explicit bound on the norm. In Sections 1.3.1–1.3.2 we will show how this can be applied to give new algorithms for sampling from fibres without needing to compute the saturation.

Theorem 1.3. *Let B be set of $n \geq 1$ vectors in $\mathbb{Z}^r$. Write β for the maximum of the absolute values of the coefficients of elements of B. Then the norm of B is at most*

$$n^n \beta^{n-1}. \tag{1.2.2}$$

Our proof (see Section 4.1) is constructive; it gives an algorithm to determine a generating set G as in the definition of the norm. We do not know whether this algorithm could be practical; it is a-priori less efficient than procedures using Gröbner bases, but is highly parallelisable.

The connection of the norm to fibre connectivity and Markov chains runs via the following result (proven in Section 4.2).

Proposition 1.4. *Let A be a $k \times r$ integer matrix, and $B = \{b_1, \ldots, b_n\}$ a basis of the kernel, with B having norm N. Let $u \in \mathbb{N}^r$, and construct a graph with vertex set the fibre $\mathcal{F}(u)$, and where we draw an edge from v_1 to v_2 if and only if $v_1 - v_2$ can be written as an integer linear combination*

$$v_1 - v_2 = \sum_{i=1}^{n} a_i b_i$$

with $\sum_{i=1}^{n} |a_i| \leq N$. Then this graph is connected.

Remark 1.5. Given a $k \times r$ integral matrix A, note that it is easy to compute a basis B of the integral kernel of A from the Smith normal form of A. Indeed, if $SAT = D$ is the Smith normal form (so S and T are invertible, and D diagonal with $D_{i,i} \mid D_{i+1,i+1}$), then let $1 \leq j \leq r$ be maximal such that $D_{j,j} \neq 0$. Then an integral basis of the kernel of A is given by $Te_{j+1}, \ldots, Te_r$, where e_i is the i-th standard basis vector in $\mathbb{Z}^r$.

Conversely, while B does not determine A, it does determine the fibres $\mathcal{F}(u)$, so the matrix A is not really essential, but is very relevant to the statistical applications.

1.2.2. *Comparison to other results in the literature.* Needless to say, we are not the first to try to control the complexity of the saturation of an ideal in a polynomial ring. Indeed, the standard method of computing the saturation reduces to a Gröbner basis computation, whose efficient implementation has been the focus of too much research to begin to list here. Specialising to the case of binomial ideals, the literature is still much too large to give more than a quick glimpse of. There are general theoretical results on the structure of fibre graphs; see, e.g., [Gross and Petrović 2013; Hemmecke and Windisch 2015; Windisch 2016; 2019]. There are also many results bounding the degree of the binomials appearing in the saturation [Haws et al. 2014; Koyama et al. 2015; Sturmfels 1996, Chapter 13], and bounding the *Markov complexity*; this is defined in [Santos and Sturmfels 2003], and studied in [Charalambous et al. 2014] and elsewhere.

However, we are not aware on bounds on the norm 1.2 in the literature. Indeed, from an algebraic point of view it appears a rather unnatural invariant. The reason for studying it comes purely from the application (via Proposition 1.4) to fibre connectivity and Markov bases. In the remainder of Section 1 we hope to justify it from this point of view, and perhaps motivate further research in this direction. An unusual feature of our results is that we do not utilise Gröbner bases; this is not from dislike, but simply because we could not see how to bound the norm from that perspective; we hope that others may have more success.

1.3. *Algorithms.*

1.3.1. *Bounded-AHT algorithms.* Aoki, Hara and Takemura connect the fibre by allowing arbitrarily large integer linear combinations of elements of the basis B. This is guaranteed to connect the fibre (since it eventually hits every integer vector), but risks wasting time searching far away from the fibre.

Proposition 1.4 shows that it actually suffices to take combinations with coefficients bounded by the norm N of B; this allows us to improve the efficiency of their algorithm, by truncating the Poisson distribution at N, spending less time exploring far from the fibre. A second algorithm they present (where the coefficients of the b_i are chosen from a multinomial distribution) can be enhanced in a similar way. An even simpler variant is to choose uniformly at random at each step a vector of L^∞-length bounded by the norm.[1]

We will refer to this class of algorithm as *bounded-AHT* algorithms, as they are characterised by the distribution used to select random vectors being of bounded support. We will see in Section 2.1 that, when a good bound on the norm is available, such an algorithm can be substantially faster than the conventional AHT algorithm.

The bound on the norm coming from Theorem 1.3 is in general large, so using it for truncation will not have a large impact on the runtime (though we hope that better bounds on the norm can be found in the future). On the other hand, if a Markov basis can be computed one can obtain a very tight bound on the norm, and our algorithm seems to converge substantially faster than that of Diaconis–Sturmfels, so it is plausible that these bounded-AHT algorithms give the best performance in these cases also.

Another application might be to predicting good values of the constant λ in the AHT algorithm, or giving heuristic bounds on the convergence time for a given value of λ. The norm N can be seen as the maximum distance between connected components of the fibre, thus to have a reasonable chance of hitting all components we should take a number of steps that is very large compared to $1/\mathbb{P}(\text{Poisson}_\lambda \geq N)$.

1.3.2. *The stepping-out algorithm.* In the naive algorithm of Section 1.1.1, one starts at a vector $v \in \mathcal{F}(u)$, and chooses at random an element $b \in \pm B$, and considers the step $v + b$. If $v + b$ is in $\mathcal{F}(u)$ then this is returned as the next element of the Markov chain. If $v + b \notin \mathcal{F}(u)$, then the algorithm simply returns v. However, if we have a bound on the norm then we can modify the algorithm so that the fibre will always be connected; if $v + b \notin \mathcal{F}(u)$ then, rather than returning v, we choose another element b_1 from $\pm B$, and consider the vector $v + b + b_1$. If $v + b + b_1$ lies in $\mathcal{F}(u)$ we return $v + b + b_1$ as the next step in the Markov chain, otherwise we repeat, until we either hit $\mathcal{F}(u)$ again, or we have taken N consecutive steps outside the fibre, in which case we return v again. Alternatively, this can be viewed as a weighted random walk in a certain graph with vertex set $\mathcal{F}(u)$. To define this graph, we first define a graph $\mathcal{F}_{\mathbb{Z}}(u)$ with vertex set $\{v \in \mathbb{Z}^r : Au = Av\}$ and with an edge between v_1 and v_2 whenever $v_1 - v_2 \in \pm B$. Then we define a graph with vertex set $\mathcal{F}(u)$ by putting an edge between two vertices whenever they can be connected by a path in $\mathcal{F}_{\mathbb{Z}}(u)$ of length at most N, and which does not intersect $\mathcal{F}(u)$ except at its endpoints. Again, by Proposition 1.4 this new graph is guaranteed to be connected.

In the examples in Section 2.1, the best performance seems to be obtained by bounded-AHT algorithms. However, we include the stepping-out algorithm because it is an example of a general technique where one can choose any algorithm to efficiently explore the interior of the fibre, and then add some "small" extra steps on the boundary to ensure that the resulting graph is connected.

1.3.3. *Speed comparisons.* In Section 2.1 we describe some numerical experiments to compare the performance of the four algorithms:

[1] That is, with the maximum entry bounded by the norm.

(1) the algorithm of Diaconis–Sturmfels using a Markov basis (DS),

(2) the algorithm of Aoki, Hara and Takemura (AHT);

(3) the bounded-AHT algorithm;

(4) the stepping-out algorithm.

The best results are obtained with the new bounded-AHT algorithm, and the worst with the DS algorithm. In between, the AHT algorithm is faster than the stepping-out algorithm. But one should not extrapolate too much from this small collection of examples.

More generally, with Theorem 1.3 and Proposition 1.4 in hand it is easy to propose new sampling algorithms which guarantee to connect the fibre. The challenge is to design algorithms with reasonable runtime, at least heuristically (rigorous runtime analysis seems hard but very interesting).

If the fibre $\mathcal{F}(u)$ is large with respect to the norm N then designing reasonably efficient algorithms is not hard, since the runtime will be dominated by time spent in the "interior" of the fibre. On the other hand, if the fibre is small compared to N then the runtime will be dominated by time spent around the edge of the fibre looking for new connected components, and will depend sensitively on the norm (or more precisely, on our bound on the norm).

1.4. *Practical consequences.*

(1) The norm bounds coming from Theorem 1.3 are in general rather large, so our new algorithms are unlikely to work very using them. We hope that these bounds can be improved, but in the meantime we note that one way to get a very good bound on the norm is simply to find a Markov basis. When this is possible it is conventional to run the algorithm of Diaconis–Sturmfels, but in Section 2.1 we illustrate that it may in fact be faster to obtain a norm bound form the Markov basis and then apply the bounded-AHT algorithm.

(2) The AHT algorithm of Section 1.1.3 is proven to converge, and in practice the Markov chain is often observed to settle down quite fast. Indeed, in practice it is the latter which will generally be relied upon; people run algorithms until the chain appears to converge. However, there is a critical problem here. Namely, we see in Section 2.2 examples where the chain will *appear* to converge very rapidly, but this "apparent" limit will not be the true limit (the runtime required to achieve true convergence may easily be arranged to exceed the lifespan of the solar system). We hope that this kind of pathological behaviour will be very rare in practice, but at present this seems hard to verify. Our aim in this paper is to get an idea of how long the algorithm should be run in order to be reasonably confident that the "apparent limit" of the chain is in fact the true limit. We are not completely successful in this, partly because our bound on the norm is rather large for practical use (and probably not sharp), and also because passing from the bound in Theorem 1.3 to an estimate on the convergence time needs substantial further work. We think it is interesting and useful to investigate this further. In the meantime, we would encourage people using this type of algorithm to let it run for as long as possible, even after the chain appears to have settled down, to maximise the change of hitting new connected components.

2. Examples

2.1. ***A very simple example.*** Consider the matrix

$$A = \begin{bmatrix} 0 & 1 & 2 & 3 \\ 3 & 2 & 1 & 0 \end{bmatrix}.$$

An integral basis for the kernel of A is then given by $B = \{b, b'\}$ where

$$b = \begin{bmatrix} 1 \\ -2 \\ 1 \\ 0 \end{bmatrix}, \quad b' = \begin{bmatrix} 0 \\ 1 \\ -2 \\ 1 \end{bmatrix}.$$

The fibre containing the vector $[2\ \ 2\ \ 2\ \ 2]^T$ is illustrated in Figure 1, where red arrows (pointing up and to the right) correspond to addition of b, and blue arrows (pointing down and to the right) to addition of b'. Evidently, this fibre is not connected, since the element $[4\ \ 0\ \ 0\ \ 4]^T$ is isolated. Thus if our chain begins anywhere in the large component it will never hit the isolated vertex, and if it begins at the isolated vertex it will remain there. This has practical consequences, since it is common to simply run such a Markov chain until it appears (by eye) to have converged; in this example, convergence will be rapid, but the resulting distribution will not be the expected one (see Section 1.4).

The approach of Diaconis–Sturmfels is to replace the basis B by a larger generating set which makes the fibre connected. The ideal $\mathcal{I}_B$ is generated by $x_1x_3 - x_2^2$ and $x_2x_4 - x_3^2$, and its saturation can be generated by these two polynomials together with the polynomial $x_1x_4 - x_2x_3$, the latter corresponding to the vector $[1\ \ -1\ \ -1\ \ 1]^T$. Clearly one can step from $[3\ \ 1\ \ 1\ \ 3]^T$ to $[4\ \ 0\ \ 0\ \ 4]^T$ by addition of this new vector, so the fibre is indeed connected by this new generating set for the integral kernel of A.

Our approach is to allow the chain to step briefly outside the fibre while it hunts for vectors with nonnegative entries. As long as we allow two negative steps the fibre will become connected, as we can step from $[3\ \ 1\ \ 1\ \ 3]^T$ to $[4\ \ 0\ \ 0\ \ 4]^T$ via $[4\ \ -1\ \ 2\ \ 3]^T$ or $[3\ \ 2\ \ -1\ \ 4]^T$; one sees easily that the norm is 2. Let us compute the bound resulting from Theorem 1.3: we have $\beta = 2$ and $n = 2$, so our bound is 8. Thus if we use the bound from the theorem we should allow 8 negative steps; it is clear that this will be sufficient to connect the fibre, but also that this bound is not optimal.

Remark 2.1. This is an opportune moment to illustrate the necessity of allowing $\epsilon_i = 0$ in Definition 1.2. In the above example the norm is 2. However, there does not exist a generating set G for the saturation of $\mathcal{I}_B$ with respect to $x_1 \cdots x_r$, with the properties that

(1) Every element of G is a pure difference binomial;

(2) Every $g \in G$ can be written in the form

$$g = \sum_{i=1}^{2} \epsilon_i m_i (x^{b_i^+} - x^{b_i^-}). \tag{2.1.1}$$

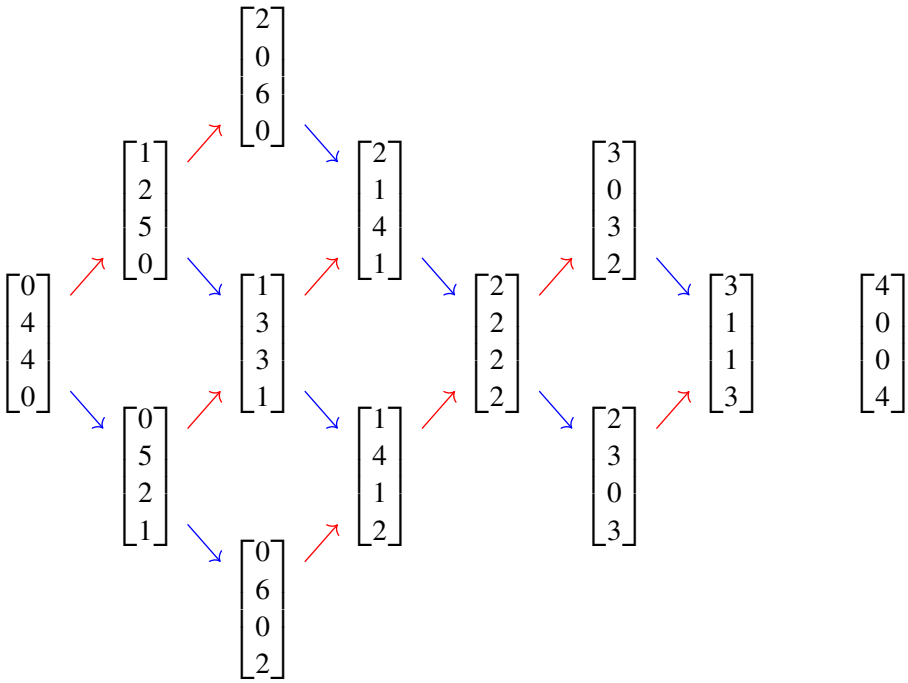


Figure 1. A (nonconnected) fibre.

where the $\epsilon_i \in \{-1, 1\}$, the m_i are Laurent monomials, and the b_i are elements of B (this differs from Definition 1.2 exactly by requiring $\epsilon_i \in \{-1, 1\}$).

To see this, suppose that G is such a generating set. Since $N = 2$ and $\#G = 2$, elementary considerations yield that every element of G is of one of the following forms:

(1) px_b where p is a polynomial consisting of two monomials with coefficients in ± 1;

(2) $px_{b'}$ where p is a polynomial consisting of two monomials with coefficients in ± 1;

(3) $m(x_1x_4 - x_2x_4)$ where m is a monomial.

We know that x_b lies in the ideal generated by G; translating into vectors, this means that b can be written as a linear combination $b = ab + a'b'$ with a, a' integer vectors whose entries sum together to an even number. This is evidently impossible.

2.2. ***Families where the fibres are arbitrarily badly connected.*** Consider the 1×3 matrix $A = [1 \;\; 1 \;\; 1]$, and write e_i for the i-th standard basis vector in $\mathbb{Z}^3$. Let $u = e_2$. Then the fibre $\mathcal{F}(u) = \{e_1, e_2, e_3\}$. For a positive integer n, choose the basis

$$B_n = \left\{ \begin{bmatrix} 0 \\ 1 \\ -1 \end{bmatrix}, \begin{bmatrix} -1 \\ n \\ 1-n \end{bmatrix} \right\}$$

of the kernel of A. Then the fibre consists of two connected components, namely $\{e_2, e_3\}$ and $\{e_1\}$. Moreover, to step between the connected components requires $(n-1)$ consecutive negative steps. Thus for every positive integer M and every real number λ there exists an integer n such that the algorithm of Aoki, Hara and Takemura presented in Section 1.1.3 applied to the above basis B_n will appear to converge immediately, but will take M steps before the probability of hitting the other connected component rises above any given positive threshold. This issue may be well-known, but this particular example appears to be new.

This example is quite artificial, as the fibre is essentially simple, but we have made a poor choice of generating set B_n. We can also construct a slightly less artificial example of the same phenomenon, by generalising the example in Section 2.1. For an integer $n \geq 2$, let

$$A_n = \begin{bmatrix} 1 & 2 & \cdots & n-1 & n \\ n & n-1 & \cdots & 2 & 1 \end{bmatrix},$$

and consider the basis of the integral kernel given by

$$B_n = \left\{ \begin{bmatrix} 1 \\ -2 \\ 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ -2 \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \cdots, \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \\ -2 \\ 1 \end{bmatrix} \right\},$$

where we denote the elements of B_n by $b_2, \ldots, b_{n-1}$ in the given order. Then the fibre of $[2 \ \cdots \ 2]^T$ contains the vector $v = [n \ \ 0 \ \cdots \ \ 0 \ \ n]^T$. This vector v is at least $n-2$ steps distant from any other point in the fibre; more precisely, if $c_1, \ldots, c_r \in \pm B_n$ are such that

$$v + \sum_{i=1}^{r} c_i \in \mathcal{F}(v),$$

then either $r \geq n-2$ or $v + \sum_{i=1}^{r} c_i = v$ (the bound $n-2$ is in fact sharp). We leave the elementary verification to the interested reader. Again we see that, though the algorithm of Section 1.1.3 (and variants) may appear to converge rapidly, there are connected components which take an arbitrarily long time to hit.

2.3. ***The no-three-factor-interaction model.*** This model is described in detail (in particular, its statistical interpretation) in [Aoki et al. 2012]. It depends on a choice of three positive integers I, J and K; we will often take $I = J = K$ for simplicity. The matrix A is then an $(IJ + JK + KI) \times IJK$ matrix, described in a slightly complicated way. Define Id_I to be the $I \times I$ identity matrix, and 1_I to be a row vector of length I with all entries equal to 1. Then

$$A = \begin{bmatrix} Id_I \otimes Id_J \otimes 1_K \\ Id_I \otimes 1_J \otimes Id_K \\ 1_I \otimes Id_J \otimes Id_K \end{bmatrix},$$

where $\otimes$ represents the Kroneker product of matrices.

Hara et al. [2012] numerically tested their algorithm 1.1.3 on the no-three-factor-interaction model in the cases $I = J = K = 3$, 5, and 10. In the case $I = 3$ the saturation can be computed by Gröbner basis techniques, but seems presently out of reach $I = 5$, and worse for $I = 10$. In each case they compute a basis for the integral kernel, then run numerical tests of their algorithm for several values of the Poisson parameter λ, and also occasionally replacing the Poisson with a different distribution (we are not completely clear on how they chose these parameters and distributions). In the case $I = 3$ they compare their results to those obtained using a saturated basis, and observe that the Markov chains coming from their algorithm converge similarly to those coming from a saturated basis (though for $\lambda = 50$ the convergence is rather slow).

For $I = 10$ their algorithm does not converge well, but for $I = 5$ it appears to converge fairly rapidly. As throughout this paper, the question we are interested in is whether this apparent convergence can be trusted, or is it possible that there is some connected component of the fibre which their chain has never hit? Of course, their algorithm will find every component with probability 1 if allowed to run for unlimited time, but there is no a-priori reason to assume that the time required for this will be in any way comparable to the time required for the chain to appear to settle down.

To try to get a handle on this, let us compute our upper bound on the number of negative steps required to walk between components (the "distance between" connected components of the fibre). Using SAGE we compute the smith normal form of the 75×125 matrix A, obtaining an integral basis B with $n = 64$ elements. The largest absolute value of an entry in B is $\beta = 1$. This leads to an upper bound on the norm by

$$N' = n^n \beta^{n-1} = 64^{64} \approx 3.9 \times 10^{115}. \tag{2.3.1}$$

Now, in this example Aoki, Hara and Takemura replace the Poisson distribution with a geometric distribution (for reasons which are unclear to us), and try parameters $p = 0.1$, 0.5. The proportion of steps in their algorithm which will exceed N' in length is then so small that it is likely never to occur before the sun runs cold. This means that *if* the bound N' were to be close to the true norm, then this algorithm will in practice never converge to the correct solution. In practice, our bound on the norm is surely very far from sharp, but we gave this example to illustrate the difficulty in guaranteeing convergence (despite the fact that the algorithm might appear to the human eye to have converged).

3. Computational experiments

3.1. ***The very simple example.*** For the example in Section 2.1 we implemented four algorithms:

(1) The algorithm of [Diaconis and Sturmfels 1998] using the Markov basis described in Section 2.1 (we refer to this algorithm as *DS*);

(2) The algorithm of Aoki, Hara and Takemura described in Section 1.1.3 (referred to as *AHT*);

(3) The bounded-AHT algorithm described in Section 1.3.1, generating vectors uniformly at random of L^∞-length up to some integer at least the norm;

(4) The stepping-out algorithm of Section 1.3.2,

so that we could compare their results. We considered the fibre containing the vector $[10, 10, 10, 10]^T$ which has 211 elements, as this small example allowed us to run many simulations to get reasonably accurate timings of the algorithms.

We use the Kolmogorov Smirnov statistic to decide how well a chain has converged. For a chain of length n in the fibre $\mathcal{F} := \mathcal{F}([10, 10, 10, 10]^T)$, a perfectly uniform distribution would sample each point $n/211$ times. Given a function $d\colon \mathcal{F} \to \mathbb{Z}$ with $\sum_{v\in\mathcal{F}} d(v) = n$ we define

$$\mathrm{KS}(d) = \max_{v\in\mathcal{F}} |d(v) - \frac{n}{211}|, \tag{3.1.1}$$

so a larger value of $\mathrm{KS}(d)$ indicates that d is further from being uniform.

There are two subtleties to comparing the outputs of the algorithms:

(1) The AHT, bounded-AHT and stepping-out algorithms have parameters that can be tuned: the mean of the Poisson distribution for AHT and the bound on the norm used in the latter two (even when the norm is known, as in this example, it is not obvious that using it as the bound will yield the best convergence). To work around this, we will tune the parameters of all three algorithms to try to get the best performance out of each for our example.

(2) When comparing runtimes, counting the number of steps in the chain is not a very good measure. Each step in AHT requires repeated sampling from a Poisson distribution, and steps in the stepping-out algorithm can involve a number of substeps outside the fibre. Because of this, we will also compare the actual runtimes, though this is then sensitive to implementation issues.

We produce a chain of $n = 211,000$ samples, so that each site expects 1000 samples. Table 1 compares the Kolmogorov Smirnov statistic and runtime for the four algorithms, making optimised choices of parameters for the AHT, bounded-AHT and stepping-out algorithms. Table 2, left, shows how the Kolmogorov Smirnov statistic and runtime for the bounded-AHT algorithm vary with the bound used, and Table 2, right, shows the same for the stepping-out algorithm.

algorithm	DS	AHT	bounded-AHT	stepping-out
KS statistic (*lower is better*)	309	210.9	162.8	270.5
runtime (s) (*lower is better*)	174	161.7	98.3	156.1
optimised parameter	-	$\lambda = 2$	$N = 3$	$N = 4$

Table 1. Comparison of algorithms (averaged over 20 runs).

Norm bound	2	3	8
KS statistic	177.3	162.8	201.4
runtime (s)	98.8	98.3	96.0

Norm bound	2	4	8
KS statistic	360.8	270.5	305.0
runtime (s)	147.6	156.1	174.3

Table 2. Comparison of norm bounds for bounded-AHT (left) and stepping-out (right), averaged over 20 runs.

We make a number of comments on these results; all come with the serious caveat that this is only a small, simple example.

(1) The improvement obtained by using the bounded-AHT algorithm in place of the original DS algorithm is quite substantial; both the KS statistic and runtime are close to being halved. This suggests that the bounded-AHT algorithm is worth investigating even in cases where a Markov basis can be computed.

(2) While bounded-AHT performs best with a norm bound $N = 3$, its performance with the bound of $N = 8$ coming from Theorem 1.3 is still better than any of the other algorithms.

(3) The worst performance is achieved by the DS algorithm (using a Markov basis), perhaps somewhat surprisingly. Even though this algorithm should explore the boundary of the fibre in a more efficient way, it probably looses out by exploring the interior less efficiently.

3.2. ***The no-three-factor-interaction model.*** Here we took $I = J = K = 5$, as this beyond the range where the saturation can currently be computed, and hence it is interesting to investigate other approaches to sampling. We implemented the stepping-out algorithm described in Section 1.3.2 for this example. Now, with the given norm bound of order 10^{115} it is clear that this algorithm will not work well. However, we remain optimistic that bounds on the norm can be improved, so it seems interesting to investigate how the runtime of the algorithm depends on the given bound. We do this in a very crude way; we simply measure the proportion of steps in the algorithm which take place within the fibre (as opposed to searching for new components outside the fibre). We interpret this as giving a very rough idea of how much slower the algorithm of Section 1.3.2 will be compared to what could be done if one had a Markov basis. The results were as follows.

(1) For a fixed fibre, when the norm bound is *large* compared to the diameter of the fibre, the runtime seems to be very roughly linear in the given bound on the norm.

(2) For a fixed fibre, when the norm bound is *small* compared to the diameter of the fibre, the runtime seems to be relatively insensitive to the size of the bound.

In other words, this might be interpreted as suggesting that the algorithm of Section 1.3.2 will work reasonably well when the norm bound is not too large compared to the diameter of the fibre.

We did not implement the bounded-AHT algorithm here; for formal reasons it is clear that it must perform slightly better than AHT, but the size of the norm bound also makes it clear that the difference will be entirely imperceptible for any practical computation.

4. Proof of the main results

4.1. ***Proof of Theorem 1.3.*** Let $B = \{b_1, \dots, b_n\}$ be a set of vectors in $\mathbb{Z}^r$. Following the notation of (1.1.2), we write

$$f_i^+ = x^{b_i^+}, \quad f_i^- = x^{b_i^-}, \quad f_i = f_i^+ - f_i^-$$

in the ring $R = \mathbb{Z}[x_1, \dots, x_r]$. Then $\mathcal{I}_B = (f_1, \dots, f_n) \subseteq R$, and our goal is to bound how far the saturation

$$\mathrm{Sat}_{x_1 \cdots x_r} \mathcal{I}_B = \{a \in R : \exists m > 0 : a(x_1 \cdots x_r)^m \in \mathcal{I}_B\} \tag{4.1.1}$$

can be from $\mathcal{I}_B$.

Definition 4.1. A *monomial* in R is an element of the form $\prod_{i=1}^r x_i^{m_i}$ with $m_i \in \mathbb{Z}_{\geq 0}$. A *pure binomial* in R is an element of the form $m_1 - m_2$ where the m_i are monomials. An ideal $I \subseteq R$ is called *pure binomial* if it admits a generating set consisting of pure binomials; evidently, $\mathcal{I}_B$ is a pure binomial ideal.

Lemma 4.2 [Herzog et al. 2018, Proposition 3.18]. *The saturation of $\mathcal{I}_B$ with respect to $x_1 \cdots x_r$ is also a pure binomial ideal.*

Definition 4.3. Given pure binomials $f = f^+ - f^-$ and $g = g^+ - g^-$, we define the *subtraction polynomial* (again a pure binomial)

$$S(f, g) = g^+ f + f^- g = f^+ g^+ - f^- g^-.$$

If $f, g \in \mathcal{I}_B$ then clearly $S(f, g)$ lies in $\mathcal{I}_B$.

We make the unsurprising notational conventions that $-- = +$, $+- = -+ = -$ and $++ = +$; thus we interpret $f^{--} = f^+$, which is less usual, but makes for efficient and hopefully comprehensible notation in what follows.

Definition 4.4. Let $\epsilon \colon \{1, \dots, n\} \to \{+, -\}$, and let $t \colon \{1, \dots, n\} \to \mathbb{N}$. Define

$$S(\epsilon, t) = \prod_{i=1}^n (f_i^{\epsilon(i)})^{t(i)} - \prod_{i=1}^n (f_i^{-\epsilon(i)})^{t(i)} \in \mathcal{I}_B, \tag{4.1.2}$$

(here we use our convention that $-- = +$ when we write $f_i^{-\epsilon(i)}$).

Lemma 4.5. *Let P be a pure binomial in $\mathcal{I}_B$. Then there exist ϵ, t, and monomials m and n such that*

$$nP = mS(\epsilon, t).$$

Proof. For the purposes of the proof, we will simplify notation by assuming that for every $b_i \in B$, the element $-b_i$ also lies in B.

Let $P \in \mathcal{I}_B$ be a pure binomial. Write $P = \sum_{j=1}^k m_j f_{i_j}$, where the m_j are monomials. We can and do assume that k is chosen minimal, and we proceed by induction on k. The case $k = 1$ is trivial.

Up to harmless sign changes, there exists a j_0 such that $m_{j_0} f_{i_{j_0}}^+ = P^+$. Reordering, we may assume that $j_0 = 1$, so

$$P - m_1 f_{i_1} = \sum_{j=2}^k m_j f_{i_j}$$

is again a pure difference binomial. By the induction hypothesis there exist monomials m and n and vectors ϵ, t with

$$m \sum_{j=2}^k m_j f_{i_j} = nS(\epsilon, t).$$

Write $S(\epsilon, t) = S^+ - S^-$. Then

$$mP = nS^+ - nS^- + m_1 f_{i_1}^+ - m_1 f_{i_1}^-.$$

Since this is a binomial, up to signs we may assume without loss of generality that $nS^- = m_1 f_{i_1}^+$. We can then write

$$f_{i_1}^+ mP = n\left(f_{i_1}^+ S^+ - f_{i_1}^- S^-\right) = nS'$$

where S' is an iterated subtraction binomial of the f_i. □

Theorem 4.6. *There exist a positive integer M, functions $\epsilon_1, \dots, \epsilon_M$ and $t_1, \dots, t_M$ as in Definition 4.4, and monomials $m_1, \dots, m_M \in R$, such that*

(1) *for all $1 \le j \le M$ we have $m_j \mid S(\epsilon_j, t_j)$;*

(2)

$$\mathrm{Sat}_{x_1 \cdots x_r} \mathcal{I}_B = \left(\frac{S(\epsilon_j, t_j)}{m_j} : 1 \le j \le M \right).$$

Proof. Combine Lemma 4.2 and Lemma 4.5. □

Given $t\colon \{1, \dots, n\} \to \mathbb{N}$ we define the L^1-length of t to be the sum of its values. To prove Theorem 1.3 it suffices to show that we can choose each of the vectors t_j in Theorem 4.6 to have L^1-length bounded by $N = n^n \beta^{n-1}$, where β is the maximum of the absolute values of entries of vectors in B; compare (1.2.2)). Given vectors ϵ of signs and t of natural numbers as in Definition 4.4, observe that the power of x_j dividing $S(t, \epsilon)$ is given by

$$\min\left(\sum_{i=1}^n t(i) \operatorname{ord}_{x_j} f_i^{\epsilon(i)}, \sum_{i=1}^n t(i) \operatorname{ord}_{x_j} f_i^{-\epsilon(i)} \right); \tag{4.1.3}$$

here $\operatorname{ord}_x f$ denotes the largest power of x which divides f. We say the *minimum in (4.1.3) is achieved on the + side* if

$$\sum_{i=1}^n t(i) \operatorname{ord}_{x_j} f_i^{\epsilon(i)} \le \sum_{i=1}^n t(i) \operatorname{ord}_{x_j} f_i^{-\epsilon(i)},$$

and we say the *minimum in (4.1.3) is achieved on the − side* if

$$\sum_{i=1}^n t(i) \operatorname{ord}_{x_j} f_i^{\epsilon(i)} \ge \sum_{i=1}^n t(i) \operatorname{ord}_{x_j} f_i^{-\epsilon(i)}.$$

Definition 4.7. Given $\epsilon\colon \{1, \dots, n\} \to \{+, -\}$ and $\delta\colon \{1, \dots, r\} \to \{+, -\}$, we define

$$T_{\epsilon,\delta} = \{t \in \mathbb{N}^n : \forall 1 \le i \le r, \text{ the minimum in (4.1.3) is achieved on the } \delta(i) \text{ side}\}.$$

This set $T_{\epsilon,\delta}$ is a rational polyhedral cone in $\mathbb{N}^n$, and for fixed ϵ we have

$$\bigcup_{\delta} T_{\epsilon,\delta} = \mathbb{N}^n. \tag{4.1.4}$$

Given $t \in T_{\epsilon,\delta}$, we write

$$\varphi_t = \frac{S(\epsilon, t)}{\prod_{j=1}^{r} x_j^{\sum_{i=1}^{n} t(i) \operatorname{ord}_{x_j} f_i^{\epsilon(i)\delta(i)}}}, \tag{4.1.5}$$

which we write as a difference of monomials $\varphi_t = \varphi_t^+ - \varphi_t^-$ in the usual way. From the definition of $T_{\epsilon,\delta}$ we see that $\varphi_t \in R$, i.e. all exponents of the x_i are nonnegative.

Lemma 4.8. *Fix ϵ and δ as above, and let $t, t_1, \ldots, t_a \in T_{\epsilon,\delta}$ such that $t = t_1 + \cdots + t_a$. Then*

$$\varphi_t \in (\varphi_{t_1}, \ldots, \varphi_{t_a}) \subseteq R.$$

Proof. Elementary manipulations yield

$$\varphi_t = \prod_{\alpha=1}^{a} \varphi_{t_\alpha}^+ - \prod_{\alpha=1}^{a} \varphi_{t_\alpha}^- = S(\cdots S(S(\varphi_{t_1}, \varphi_{t_2})\varphi_{t_3}) \cdots \varphi_{t_a}). \qquad \square$$

Theorem 4.9. *For each ϵ and each δ, choose a generating set $\tau_{\epsilon,\delta}$ for the cone $T_{\epsilon,\delta}$. Then*

$$\bigcup_{\epsilon,\delta} \{\varphi_t : t \in \tau_{\epsilon,\delta}\} \tag{4.1.6}$$

is a generating set for $\operatorname{Sat}_{x_1 \cdots x_r} \mathcal{I}_B$.

Proof. Let $t \in \mathbb{N}^n$, then $S(\epsilon, t) \in \mathcal{I}_B$, and $\varphi_t \in R$, hence by definition of the saturation we see that $\varphi_t \in \operatorname{Sat}_{x_1 \cdots x_r} \mathcal{I}_B$. Conversely, Theorem 4.6 tells us that the φ_t generate $\operatorname{Sat}_{x_1 \cdots x_r} \mathcal{I}_B$ as t ranges over $\mathbb{N}^n$. We must justify why it suffices to consider only t ranging over the set in (4.1.6). Fixing ϵ, we note that every $t \in \mathbb{N}^r$ lies in some $T_{\epsilon,\delta}$ by (4.1.4), and then by Lemma 4.8 it suffices to range over elements of a generating set for $T_{\epsilon,\delta}$. $\square$

Fixing ϵ and δ, it remains to show that $T_{\epsilon,\delta}$ can be generated by vectors of length bounded by $N = n^n \beta^{n-1}$. First, we have an elementary lemma.

Lemma 4.10. *Let $v_1, \ldots, v_a \in \mathbb{N}^n$, and let C be the intersection of $\mathbb{N}^n$ with the* rational *cone spanned by the v_i. Then C is generated by*

$$C \cap \left\{ \sum_{i=1}^{a} \lambda_i v_i : \lambda_i \in [0, 1) \right\} \cup \{v_1, \ldots, v_a\}.$$

Observe that the faces of $T_{\epsilon,\delta}$ are defined by the equations

$$\sum_{i=1}^{n} t(i) \operatorname{ord}_{x_j} f_i^{\epsilon(i)} = \sum_{i=1}^{n} t(i) \operatorname{ord}_{x_j} f_i^{-\epsilon(i)}; \tag{4.1.7}$$

thus the extremal rays of $T_{\epsilon,\delta}$ are obtained by solving $n-1$ equations of the form (4.1.7). Let β be the maximum of the absolute values of the $\operatorname{ord}_{x_j} f_i = b_{i,j}$ as i and j vary. Observing that for any given i and j at least one of $\operatorname{ord}_{x_j} f_i^{\epsilon(i)}$ and $\operatorname{ord}_{x_j} f_i^{-\epsilon(i)}$ is equal to zero, we can rearrange these equations to the form $\sum_i \beta_{i,j,\epsilon} t(i) = 0$ with $\beta_{i,j,\epsilon}$ an integer of absolute value not greater than β. By Siegel's lemma,

the L^1-length of such a (nonzero) solution is then bounded above by $(n\beta)^{n-1}$. From Lemma 4.10, and cutting into simplicial cones, we see that $T_{\epsilon,\delta}$ can be generated by vectors of length at most $N = n^n\beta^{n-1}$, concluding the proof.

4.1.1. *Detailed description of the $T_{\epsilon,\delta}$.* The $T_{\epsilon,\delta}$ for fixed ϵ and varying δ resemble the cones of a complete polyhedral fan in $\mathbb{N}^n$ in the sense of [Fulton 1993]. More precisely, they form a collection of polyhedral cones in $\mathbb{N}^n$ which cover $\mathbb{N}^n$ and such that the intersection of any two cones is a face of both. However, they do not quite form a fan, for two reasons:

(1) it can happen that $T_{\epsilon,\delta} = T_{\epsilon,\delta'}$ for $\delta \neq \delta'$;

(2) the intersection of two $T_{\epsilon,\delta}$ does not necessarily occur among the $T_{\epsilon,\delta}$.

However, by throwing away duplicate cones and appending the intersections of cones, one does obtain a complete fan. The corresponding toric variety is then a toric blowup of affine space $\mathbb{A}^n$.

In the example of Section 2.1 we have $n = 2$ and $r = 4$, and so the fans can readily be drawn for each ϵ. We use this to illustrate the above comments in Table 3.

To explain this in more detail for the case $\epsilon = (+,+)$ (i.e. ϵ taking the constant value $+$), the fan is obtained by subdividing $\mathbb{N}^2$ along the rays through $(1,2)$ and $(2,1)$. For each δ we describe in Table 4 the fan $T_{(+,+),\delta}$.

Our main work in this proof is to bound the lengths of generators for these cones. The general bound we obtained is $N = n^n\beta^{n-1}$, which in this case yields $N = 8$. However, just from studying the last row of Table 3 we see that we can take a generating set to be

$$(1,0),\ (0,1),\ (1,2),\ (2,1),\ (1,1). \tag{4.1.8}$$

In particular, we obtain a bound on the norm of 3. This is very close to sharp, as we saw in Section 2.1 that the norm is 2. This illustrates that a major source of nonsharpness in our bound is the application of Siegel's lemma below. It seems reasonable to hope that one can find better bounds on the norm by studying Hilbert bases for the cones $T_{\epsilon,\delta}$.

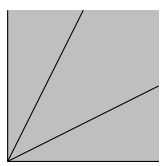
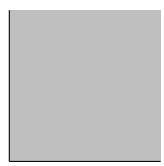
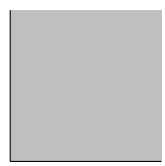
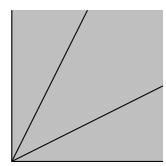
ϵ	$(+,+)$	$(+,-)$	$(-,+)$	$(-,-)$
rays generating fan	(1,2), (2,1)	-	-	(1,2), (2,1)
fan				
generating set for all cones	(0,1), (1,0), (1,2), (2,1), (1,1)	(1,0), (0,1)	(1,0), (0,1)	(0,1), (1,0), (1,2), (2,1), (1,1)

Table 3. The fans generated by the $T_{\epsilon,\delta}$.

$\delta(1)\delta(2)\delta(3)\delta(4)$	cone
$++++$	$\{(0,0)\}$
$+++-$	$\{(0,0)\}$
$++-+$	$\{(0,0)\}$
$++--$	$\{(0,0)\}$
$+-++$	$\{(0,0)\}$
$+-+-$	$\langle(0,1)\rangle$
$+--+$	$\{(0,0)\}$
$+---$	$\{(0,0)\}$
$-+++$	$\{(0,0)\}$
$-++-$	$\langle(1,2),(2,1)\rangle$
$-+-+$	$\langle(1,0)\rangle$
$-+--$	$\langle(0,1),(1,2)\rangle$
$--++$	$\{(0,0)\}$
$--+-$	$\langle(1,0),(2,1)\rangle$
$---+$	$\{(0,0)\}$
$----$	$\{(0,0)\}$

Table 4. The cones $T_{(++),\delta}$.

4.2. ***Proof of Proposition 1.4.*** Let G be a generating set for the saturation as in Definition 1.2. Each $g \in G$ is a pure difference binomial, say $g = x^{c^+} - x^{c^-}$ with $c^+, c^- \in \mathbb{N}^r$, and can be written in the form

$$g = \sum_{i=1}^{N} \epsilon_i m_i f_{j_i},$$

with $\epsilon_i \in \{1, 0, -1\}$, m_i monomials, and f_j as in Section 4.1. Writing $c = c^+ - c^-$, it suffices (by Theorem 1.1) to show that c can be written as $c = \sum_{i=1}^{n} a_i b_i$ with $\sum_{i=1}^{n} |a_i| \le N$.

We wish to prove this by induction on N, but this makes no sense as N is the norm. Instead we rephrase things slightly so that induction makes sense, resulting in the following lemma.

Lemma 4.11. *Let M be a positive integer, and suppose that the expression*

$$\sum_{i=1}^{M} \epsilon_i m_i f_{j_i}, \tag{4.2.1}$$

is a pure binomial $x^{c^+} - x^{c^-}$, where $\epsilon_i \in \{1, -1\}$, and the m_i are monomials. Then there exist integers $a_1, \ldots, a_n$ with $\sum_{i=1}^{n} |a_i| \le M$ and $c^+ - c^- = \sum_{i=1}^{n} a_i b_i$.

It is clear that the lemma (applied with $M = N$) implies Proposition 1.4, so it only remains to verify the lemma.

Proof. For a warmup we treat first the case $M = 1$. Then

$$x^{c^+} - x^{c^-} = \pm m(x^{b^+_{j_1}} - x^{b^-_{j_1}}) = \pm(x^{d+b^+_{j_1}} - x^{d+b^-_{j_1}})$$

where we write $m = x^d$ for some $d \in \mathbb{N}^r$. Hence

$$c^+ - c^- = \pm((d + b_{j_1}^+) - (d + b_{j_1}^-)) = \pm b_{j_1}$$

as required.

We prove the general case by induction on M. First, up to changing some signs, observe that we can reorder the terms in the expression (4.2.1) so that $m_M f_{j_M}^+ = x^{c^+}$, hence we can assume that $\sum_{i=1}^{M-1} \epsilon_i m_i f_{j_i}$ is also a pure binomial, say

$$\sum_{i=1}^{M-1} \epsilon_i m_i f_{j_i} = x^{c'^+} - x^{c'^-}.$$

By our induction hypothesis we can write $c'^+ - c'^- = \sum_{i=1}^n a_i' b_i$ with $\sum_{i=1}^n |a_i'| \leq M - 1$. Then

$$\sum_{i=1}^{M-1} \epsilon_i m_i f_{j_i} = x^{c'^+} - x^{c'^-} = x^{c^+} - x^{c^-} - \epsilon_M m_M (x^{b_{j_M}^+} - x^{b_{j_M}^-}),$$

and we can (again changing some signs, without loss of generality) assume that $\epsilon_M = +1$ and that $x^{c^-} = m_M x^{b_{j_M}^+}$. Writing $m_M = x^d$, we see

- $x^{c^+} = x^{c'^+}$, so $c^+ = c'^+$;
- $x^{c^-} = x^{d + b_{j_M}^+}$, so $c^- = d + b_{j_M}^+$;
- $x^{c'^-} = m_M x^{b_{j_M}^-} = x^{d + b_{j_M}^-}$, so $c'^- = d + b_{j_M}^-$.

Putting these together we see

$$c^+ - c^- = c'^+ - c^- = (c'^+ - c'^-) + (b_{j_M}^+ - b_{j_M}^-) = (c'^+ - c'^-) + b_{j_M},$$

from which the result is immediate. □

Acknowledgements

This work owes its existence to a seminar on algebraic statistics organised in Leiden in the Autumn of 2018 by Garnet Akeyr, Rianne de Heide, and Rosa Winter. I am very grateful to them for organising it, for the many expert speakers who took the time to patiently explain basic ideas of probability and statistics to us, and especially to the determined participants who survived to the end, and offered very useful comments on a presentation of the results contained here.

I am also grateful to the referees for encouraging me to improve the paper, and to the editors for their graceful handling of the apoptosis and apocatastasis of the journal.

References

[Aoki et al. 2012] S. Aoki, H. Hara, and A. Takemura, *Markov bases in algebraic statistics*, Springer, 2012.

[Bigatti et al. 1999] A. M. Bigatti, R. La Scala, and L. Robbiano, "Computing toric ideals", *J. Symbolic Comput.* **27**:4 (1999), 351–365.

[Bosma et al. 1997] W. Bosma, J. Cannon, and C. Playoust, "The Magma algebra system, I: The user language", *J. Symbolic Comput.* **24**:3-4 (1997), 235–265.

[Charalambous et al. 2014] H. Charalambous, A. Thoma, and M. Vladoiu, "Markov complexity of monomial curves", *J. Algebra* **417** (2014), 391–411.

[Decker et al. 2019] W. Decker, G.-M. Greuel, G. Pfister, and H. Schönemann, "Singular 4-1-2 — a computer algebra system for polynomial computations", 2019, available at http://www.singular.uni-kl.de.

[Diaconis and Sturmfels 1998] P. Diaconis and B. Sturmfels, "Algebraic algorithms for sampling from conditional distributions", *Ann. Statist.* **26**:1 (1998), 363–397.

[Fulton 1993] W. Fulton, *Introduction to toric varieties*, Annals of Mathematics Studies **131**, Princeton University Press, 1993.

[Gross and Petrović 2013] E. Gross and S. Petrović, "Combinatorial degree bound for toric ideals of hypergraphs", *Internat. J. Algebra Comput.* **23**:6 (2013), 1503–1520.

[Hara et al. 2012] H. Hara, S. Aoki, and A. Takemura, "Running Markov chain without Markov basis", pp. 45–62 in *Harmony of Gröbner bases and the modern industrial society*, edited by T. Hibi, World Sci., Hackensack, NJ, 2012.

[Haws et al. 2014] D. Haws, A. Martín del Campo, A. Takemura, and R. Yoshida, "Markov degree of the three-state toric homogeneous Markov chain model", *Beitr. Algebra Geom.* **55**:1 (2014), 161–188.

[Hemmecke and Windisch 2015] R. Hemmecke and T. Windisch, "On the connectivity of fiber graphs", *J. Algebr. Stat.* **6**:1 (2015), 24–45.

[Hemmecke et al. 2001–] R. Hemmecke, R. Hemmecke, M. Köppe, P. Malkin, and M. Walter, "4ti2—a software package for algebraic, geometric and combinatorial problems on linear spaces", 2001–, available at https://4ti2.github.io/.

[Herzog et al. 2018] J. Herzog, T. Hibi, and H. Ohsugi, *Binomial ideals*, Graduate Texts in Mathematics **279**, Springer, 2018.

[Koyama et al. 2015] T. Koyama, M. Ogawa, and A. Takemura, "Markov degree of configurations defined by fibers of a configuration", *J. Algebr. Stat.* **6**:2 (2015), 80–107.

[Santos and Sturmfels 2003] F. Santos and B. Sturmfels, "Higher Lawrence configurations", *J. Combin. Theory Ser. A* **103**:1 (2003), 151–164.

[Sturmfels 1996] B. Sturmfels, *Gröbner bases and convex polytopes*, University Lecture Series **8**, American Mathematical Society, Providence, RI, 1996.

[Windisch 2016] T. Windisch, "Rapid mixing and Markov bases", *SIAM J. Discrete Math.* **30**:4 (2016), 2130–2145.

[Windisch 2019] T. Windisch, "The fiber dimension of a graph", *Discrete Math.* **342**:1 (2019), 168–177.

Received 2019-09-17. Revised 2020-05-26. Accepted 2020-07-06.

DAVID HOLMES: holmesdst@math.leidenuniv.nl
, *Mathematisch Instituut Leiden, Niels Bohrweg 1, 2333 CA Leiden, Netherlands*